

Trabalho Prático de Técnicas de construção de programas

Anderson Castilhos

João Vicente Lessa

Mauricio Rocha

Linguagem Utilizada

- Linguagem utilizada: Java.
- Motivos: É uma linguagem robusta e disseminada, com recursos que facilitariam o trabalho.
- Framework utilizado: Jplay.





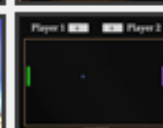
O framework JPlay foi construído tendo como objetivo ser de fácil utilização por parte de alunos que estão iniciando no aprendizado de programação. Sendo uma ferramenta que possibilite ao aluno um modo fácil de desenhar imagens na tela do computador e movimentá-las. Com vários métodos e objetos auxiliares que o ajudarão a criar jogos 2D usando a linguagem Java em pouquíssimo tempo.

Ao lado, na galeria de imagens, estão alguns dos jogos feitos por alunos da UFF.

Obs.: A IDE usada foi NetBeans 6.8 e a versão do JDK é a 1.6.

Não deixe de olhar a parte de extensões, há funcionalidades interessantes que podem ser adicionados ao seu jogo.

Galeria de Imagens





Package jplay


Class Summary	
Animation	Class responsible for animating a GameImage using pieces of the image, such as frames.
Collision	Class used to know whether two GameObjects collided.
GameImage	Class responsible for modeling an image.
GameObject	The most basic class presents in the framework.
InputBase	Class used to handle actions for buttons or for keys.
Keyboard	Class responsible for handling the keys of keyboard and its behavior.
Mouse	Class responsible for handling mouse actions.
Scene	Class responsible for handling a Scenario.
Sound	Class responsible for controlling the execution of sounds.
Sprite	Class responsible for controlling all actions and behaviors of sprite.
TileInfo	Class used to handle TileInfo.
Time	Class used to manipulate time.
Window	Main class of the framework.

- Familiarização com o framework acompanhou o desenvolvimento inicial do jogo .
- Projeto do jogo levando em contas os recursos oferecidos

INTERFACE

 New Game

 Scores

 Exit Game



Time Left:

Level:

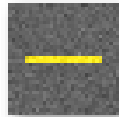
Enter and Arrows to Select Esc to go Back

O Mapa

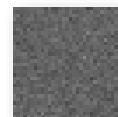
- É uma instância da classe Scene
- Composto por 800x576 pixels
- Organizado em uma matriz de tiles 25x18
 - Cada tile possui 32x32 pixels
- Tiles:



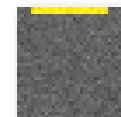
asfalto.png



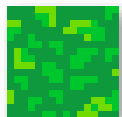
asphalt01.png



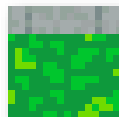
asphalt02.png



asphalt03.png



grass.png



grassborder.png



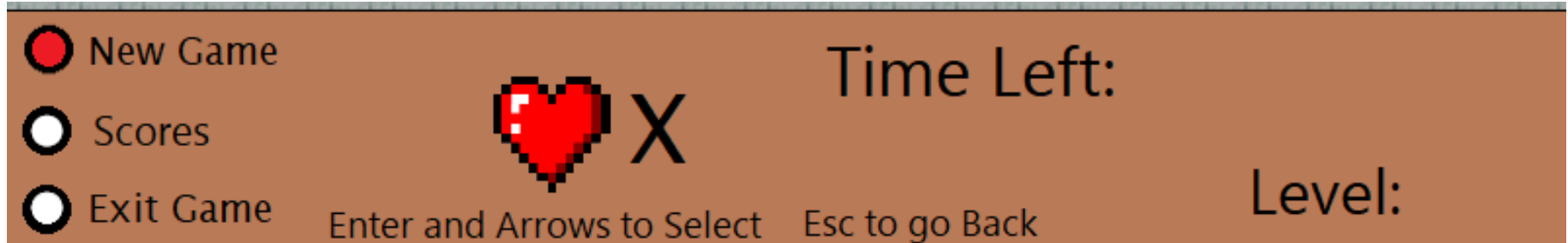
grassborder02.png



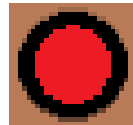
wood.png

Menu

Fundo estático e imagens sobrepostas



Indicador de seleção:



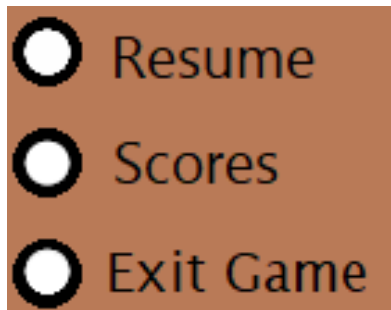
Contador de vidas:



Timer (Jplay):

Time left: 00:02:51

Menu de Pause:



Alertas:



Classes herdeiras de sprite

```
public class Frog extends Sprite {  
    public Frog(double x, double y)  
    public boolean atravessouAvenida()  
    public void avoidWalls(Scene scene)  
    public boolean vehicleColision(Vehicle vehicle)  
    public void move(Keyboard keyboard, Window window)  
    private boolean tileColision(GameObject obj, TileInfo tile)  
    private boolean verticalColision(GameObject obj, GameObject vehicle)  
    public void toStartPosition()  
    ...  
}
```



Classes herdeiras de sprite

```
public abstract class Vehicle extends Sprite
public void move()
public void accelerate()
public double getSpeed()
public void setSpeed(double speed)
```



```
public class SpecialCar extends Vehicle  
// @Override  
public void move(double frog_y, int pista_y)
```

Diagrama de Classes

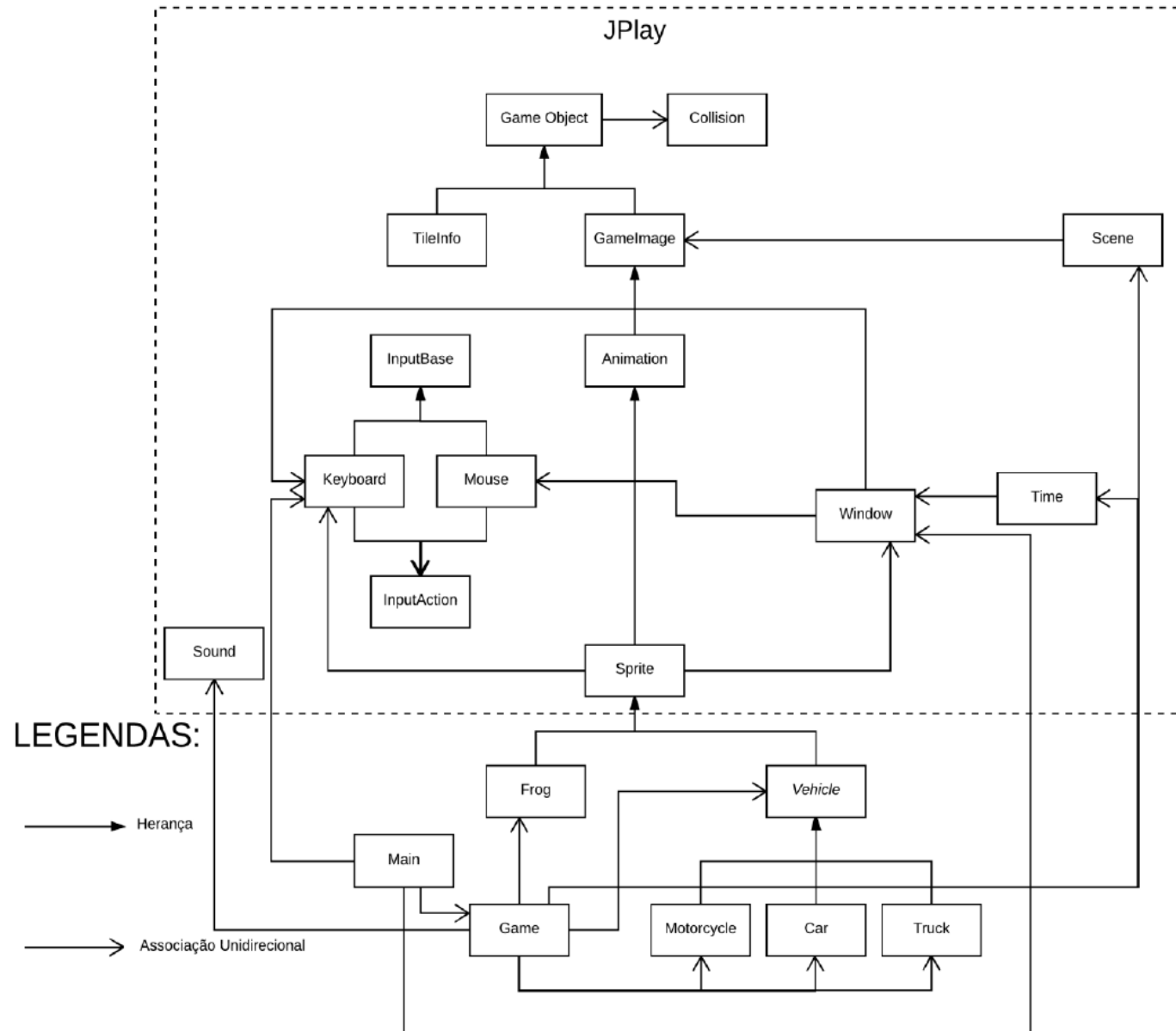
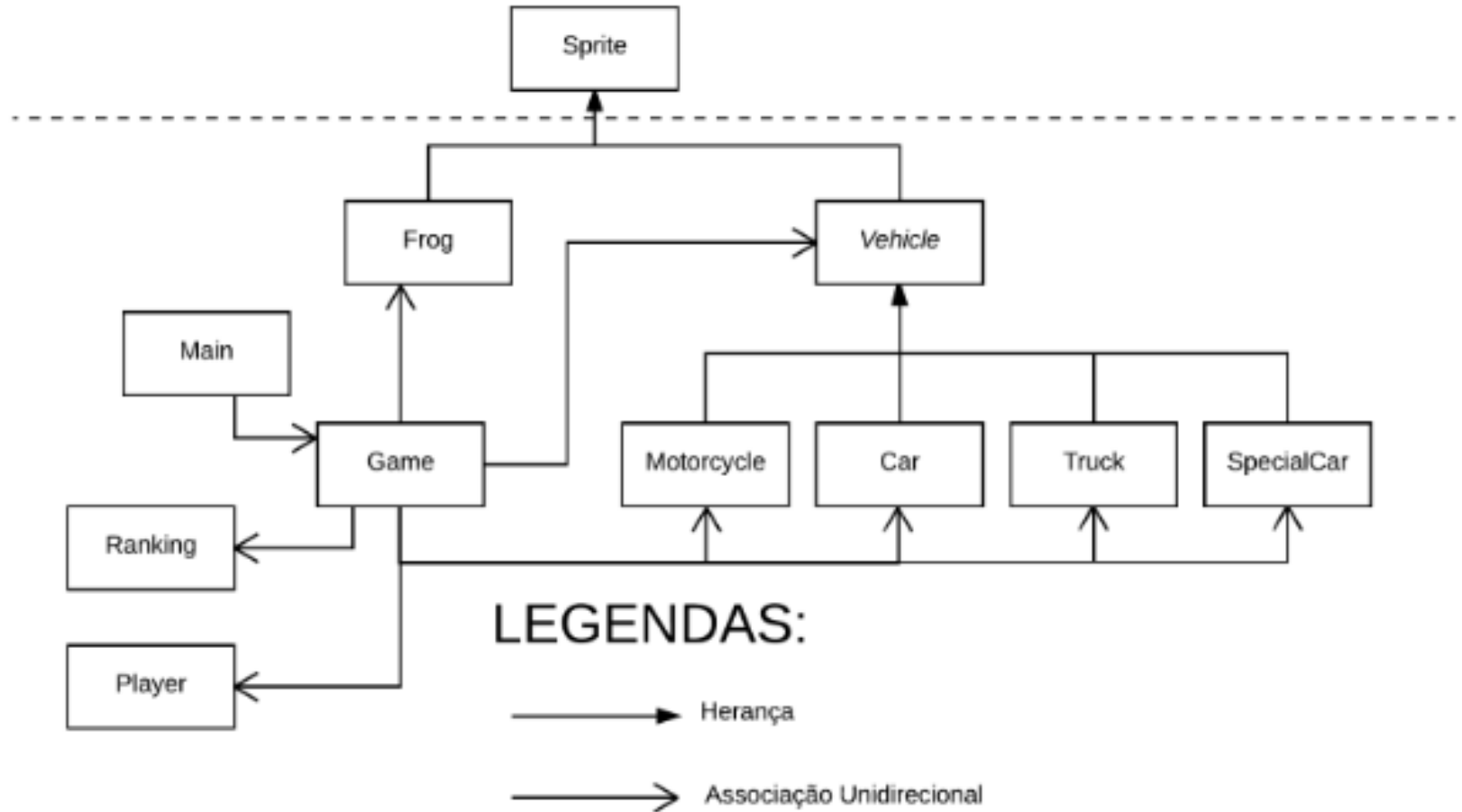


Diagrama de Classes



Ranking

- Lê do arquivo e armazena 11 nomes e as respectivas pontuações em dois arrays, mostra só os 10 primeiros
- Armazenado em um arquivo txt seguindo o seguinte formato

```
1 <defaultName>
2 0
3 <defaultName>
4 0
5 <defaultName>
6 0
7 <defaultName>
8 0
9 <defaultName>
10 0
11 <defaultName>
12 0
13 <defaultName>
14 0
15 <defaultName>
16 0
17 <defaultName>
18 0
19 <defaultName>
20 0
21 <defaultName>
22 0
```

- Ordena pela Pontuação

```
public class Ranking
{
    int score[];
    String name[];
    public void readFile(String path) throws IOException
    public void writeFile(String path) throws IOException
    public void initializeNames()
    public void initializeScores()
    public void scoreSort()
    public void show(Window window)
}
```